WHAT IS CLAIMED IS:

1. A method for restarting a network device, the network device having a plurality of route processors, the network device being arranged to be in communication with a plurality of peers within a network, the plurality of peers including a first peer and a second peer, the first peer being arranged to support graceful restart, the method comprising:

performing a graceful restart with respect to the first peer; and

performing a peer transparent failover with respect to the second peer, wherein the graceful restart and the peer transparent failover are performed in response to downtime associated with the network device.

2. The method of claim 1 wherein when a first route processor of the plurality of route processors boots up, the method further includes:

marking a session between the network device and the second peer as a transparent failover session.

3. The method of claim 1 wherein performing the graceful restart with respect to the first peer includes reestablishing a session between the network device and the first peer using a standby route processor of the plurality of route processors.

4. The method of claim 1 wherein performing the peer transparent failover includes:

processing an event queue associated with a standby route processor of the plurality of route processors to recreate at least one state associated with a session between the network device and the second peer on the standby route processor.

5. The method of claim 4 wherein performing the peer transparent failover further includes:

identifying at least one send packet that is to be sent on the standby route processor; and

sending out a dummy packet to the second peer using the standby route processor to substantially fill in for the at least one send packet.

6. The method of claim 5 wherein performing the peer transparent failover further includes:

executing an algorithm, the algorithm being arranged to identify a path through the network; and

sending at least one update to the second peer using the path.

7. The method of claim 6 wherein performing the peer transparent failover further includes:

identifying a withdraw using a database associated with the standby route processor; and

sending the withdraw when the withdraw is identified.

8. The method of claim 7 wherein performing the peer transparent failover further includes:

substantially synchronizing the standby route processor and an active route processor of the plurality of route processors.

9. The method of claim 8 wherein substantially synchronizing the standby route processor and the active route processor includes:

identifying the standby route processor as a first route processor;

identifying the active route processor as a second route processor;

creating a connection on the first route processor;

creating a connection on the second route processor that is substantially the same as the connection on the first route processor;

receiving a packet on the first route processor;

mirroring the packet on the second route processor;

creating a set of substantially all states associated with the packet on the first route processor; and

creating a subset of the set of substantially all states on the second route processor.

10. The method of claim 9 wherein the subset of the set of substantially all states is a substantially minimum number of required states.

11. The method of claim 1 wherein the peer transparent failover is a stateful switchover.

12. The method of claim 1 wherein the network device is a router.

13. A device suitable for use in a network, the being arranged to be in communication with a first peer and a second peer, the first peer being arranged to support graceful restart, the device comprising:

a plurality of route processors;

code devices that determine when the device is to be restarted;

code devices that cause a graceful restart to be performed with respect to the first peer when it is determined that the device is to be restarted;

code devices that cause a peer transparent failover to be performed with respect to the second peer when it is determined that the device is to be restarted; and

a medium that stores the code devices.

14. The device of claim 13 further including:

code devices that cause a session between the network device and the second peer to be identified as a transparent failover session when a first route processor of the plurality of route processors boots up.

15. The device of claim 13 wherein the code devices that cause the peer transparent failover to be performed include:

code devices that cause an event queue associated with a standby route processor of the plurality of route processors to be processed to recreate at least one state associated with a session between the device and the second peer on the standby route processor.

16. The device of claim 15 wherein the code devices that cause the peer transparent failover to be performed further include:

code devices that cause at least one send packet that is to be sent on the standby route processor to be identified; and

code devices that cause a dummy packet to be sent to the second peer using the standby route processor to substantially fill in for the at least one send packet.

17. The device of claim 16 wherein the code devices that cause the peer transparent failover to be performed further include:

code devices that cause an algorithm to be executed, wherein the algorithm is arranged to identify a path through a network that includes the device; and

code devices that cause at least one update to be sent to the second peer using the path.

18. The device of claim 17 wherein the code devices that cause the peer transparent failover to be performed further include:

code devices that cause the standby route processor and an active route processor of the plurality of route processors to be substantially synchronized.

19. The device of claim 18 wherein the code devices that cause the standby route processor and the active route processor to be substantially synchronized include:

code devices that cause the standby route processor to be identified as a first route processor;

code devices that cause the active route processor to be identified as a second route processor;

code devices that cause a connection to be created on the first route processor;

code devices that cause a connection to be created on the second route processor that is substantially the same as the connection on the first route processor;

code devices that cause a packet to be received on the first route processor;

code devices that cause the packet to be mirrored on the second route processor;

code devices that cause a set of substantially all states associated with the packet on the first route processor to be created; and

code devices that cause a subset of the set of substantially all states to be created on the second route processor.

20. The device of claim 19 wherein the subset of the set of substantially all states is a substantially minimum number of required states.

21. The device of claim 13 wherein the peer transparent failover is a stateful switchover.

22. The device of claim 13 wherein the device is a router.

23. A device suitable for use in a network, the being arranged to be in communication with a first peer and a second peer, the first peer being arranged to support graceful restart, the device comprising:

a plurality of route processors;

means for performing a graceful restart with respect to the first peer; and

means for performing a peer transparent failover with respect to the second peer, wherein the graceful restart and the peer transparent failover are performed in response to a resetting process associated with the device.

24. The device of claim 23 further including:

means for identifying a session between the network device and the second peer as a transparent failover session when a first route processor of the plurality of route processors boots up.

25. The device of claim 23 wherein the means for performing the graceful restart with respect to the first peer include means for reestablishing a session between the network device and the first peer using a standby route processor of the plurality of route processors.

26. The device of claim 23 wherein the means for performing the peer transparent failover include:

means for processing an event queue associated with a standby route processor of the plurality of route processors to recreate at least one state associated with a session between the device and the second peer on the standby route processor.

27. The device of claim 26 wherein the means for performing the peer transparent failover further include:

means for identifying at least one send packet that is to be sent on the standby route processor; and

means for sending out a dummy packet to the second peer using the standby route processor to substantially fill in for the at least one send packet.

28. The device of claim 27 wherein the means for performing the peer transparent failover further include:

means for executing an algorithm, the algorithm being arranged to identify a path through the network; and

means for sending at least one update to the second peer using the path.

29. The device of claim 28 wherein the means for performing the peer transparent failover further include:

means for substantially synchronizing the standby route processor and an active route processor of the plurality of route processors.

30. The device of claim 23 wherein the peer transparent failover is a stateful switchover.

31. A method for recovering a routing state associated with a network device when the network device is reset, the network device having a plurality of route processors, the network device being arranged to be in communication with a plurality of peers within a network, the plurality of peers including a first peer and a second peer, the first peer being arranged to support graceful restart, the method comprising:

performing a graceful restart with respect to the first peer; and

performing a peer transparent failover with respect to the second peer, wherein performing the peer transparent failover includes processing an event queue maintained on a standby route processor of the plurality of route processors to substantially recreate the routing state, wherein the graceful restart and the peer transparent failover are performed when a connection associated with the network device is reset.

32. The method of claim 31 wherein the peer transparent failover is a stateful switchover.

33. A device suitable for use in a network, the being arranged to be in communication with a first peer and a second peer, the first peer being arranged to support graceful restart, the device comprising:

a plurality of route processors;

a graceful restart arrangement arranged to perform a graceful restart with respect to the first peer; and

a peer transparent failover arrangement arranged to perform a peer transparent failover with respect to the second peer, wherein the graceful restart and the peer transparent failover are performed in response to a resetting process associated with the device.

34. The device of claim 33 further including:

a session identifier, the session identifier being arranged to identify a session between the network device and the second peer as a transparent failover session when a first route processor of the plurality of route processors boots up.

35. The device of claim 33 wherein the graceful restart arrangement arranged to performing the graceful restart with respect to the first peer includes a session reestablishment arrangement arranged to reestablish a session between the network device and the first peer using a standby route processor of the plurality of route processors.